(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
6 June 2002 (06.06.2002)

PCT

(10) International Publication Number
WO 02/45375 A2

(51) International Patent Classification[7]: H04L 29/06, 12/56, 12/46

(21) International Application Number: PCT/EP01/14203

(22) International Filing Date: 29 November 2001 (29.11.2001)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
60/250,630 1 December 2000 (01.12.2000) US
60/276,059 15 March 2001 (15.03.2001) US

(71) Applicant *(for all designated States except US)*: **NORTEL NETWORKS LIMITED** [CA/CA]; 2351 Boulevard Alfred-Nobel, St. Laurent, Quebec H4S 2A9 (CA).

(72) Inventors; and

(75) Inventors/Applicants *(for US only)*: **CHRISTIAN, Philip** [GB/GB]; 12 Ardley Crescent, Hatfield Heath, Bishop's Stortford, Hertfordshire CM22 7AH (GB). **FEE, Paul** [GB/GB]; 36 Mountjoy Road, Dungannon, BT71 5DH (GB).

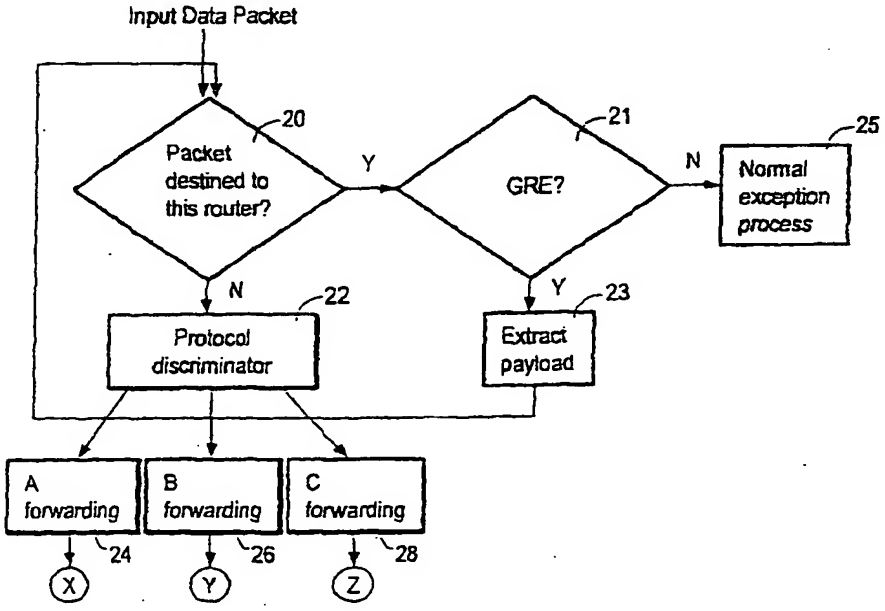
(74) Agent: **O'CONNELL, Maura**; F.R. Kelly & Co., 9 University Street, Belfast BT7 1FY, Northern Ireland (GB).

(81) Designated States *(national)*: AE, AG, AL, AM, AT, AT (utility model), AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DE (utility model), DK, DK (utility model), DM, DZ, EE, EE (utility model), ES, FI, FI (utility model), GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, OM, PH, PL, PT, RO, RU, SD, SE, SG, SI, SK, SK (utility model), SL, TJ, TM, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZM, ZW.

(84) Designated States *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM),

[Continued on next page]

(54) Title: AUTO-TUNNELLING IN A HETEROGENOUS NETWORK



(57) Abstract: The invention relates to auto-tunnelling in a heterogenous network, and more particularly in an IS-IS routing domain. The preferred embodiment provides a routing apparatus that enables network nodes which support incompatible network layer protocols, such as CLNP, IPv4 or IPv6 to be present in a single routing domain, e.g. IS-IS level-1 area or level-2 subdomain. The routing apparatus enables nodes automatically to tunnel one network layer protocol over another as required, provided that all of the nodes in the routing domain support IS-IS or Integrated IS-IS routing protocols. The routing apparatus does not necessarily try to route a data packet to its destination node but rather sends the data packet in a tunnel to a node in the path to the destination node, as appropriate.

WO 02/45375 A2

European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Declarations under Rule 4.17:

— *as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii)) for the following designations AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, OM, PH, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, UZ, VN, YU, ZA, ZM, ZW, ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG)*

— *of inventorship (Rule 4.17(iv)) for US only*

Published:

— *without international search report and to be republished upon receipt of that report*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

Auto-tunnelling in a heterogeneous network

Field of the invention

The present invention relates to routing data packets in a heterogeneous network. In particular the invention relates to auto-tunnelling in a heterogeneous network, and more particularly in an IS-IS routing domain.

Background

Integrated IS-IS (Intermediate System to Intermediate System) routing protocol was devised as an extension to IS-IS routing protocol to allow it to route IP (Internet Protocol) traffic as well as OSI (Open Systems Interconnection) traffic.

Integrated IS-IS is described in IETF (Internet Engineering Task Force) standard RFC 1195 which can be obtained from http://www.ietf.org/rfc/rfc1195.txt. IS-IS is described in ISO (International Organization for Standardization) 10589 which can be obtained from http://www.iso.ch.

Integrated IS-IS has now been extended to route IP version 6 (IPv6) too. This has been described in published documents http://www.ietf.org/internet-drafts/draft-ietf-isis-ipv6-01.txt. and http://www.ietf.org/internet-drafts/draft-ietf-isis-ipv6-02.txt. Further, a derivative

of IS-IS is also used to route Digital Equipment Corporation's DECnet phase V protocol.

Integrated IS-IS is conventionally intended to be used to route data packets conforming with OSI or IP protocols simultaneously but suffers from the following rule:

From RFC 1195: "There may be times when a dual router has to forward an IP packet to an OSI-only router, or forward an OSI packet to an IP-only router. In this case the packet must be discarded." and "Similarly, due to errors, in some cases an IP-only router may have to forward an IP packet to an OSI-only router. Again, the packet must be discarded, as specified above. This may only occur if IP-only and OSI-only routers occur in the same area, which is a configuration error."

The conclusion is that IP-only and OSI-only routers cannot appear in the same IS-IS level-1 network area or in the same level-2 sub-domain. For this reason RFC 1195 [1] places topological restrictions on networks that are routed by Integrated IS-IS, requiring that all of the nodes support both IP and OSI network layer protocols in an area that have both OSI traffic and IP traffic present in them. Consequently, according to RFC 1195 [1], if one node is upgraded to support IP, then all of the others in the level-1 area or level-2 subdomain must also be upgraded.

Products do exist in the marketplace that are able to tunnel OSI data packets over IP data packets and vice

versa. These products rely on manual configuration of tunnels, and cannot break the above rule about IP only and OSI only routers appearing in the same IS-IS area or sub-domain. Further, manual tunnelling requires a lot of provisioning and may therefore be considered to be undesirable.

Also, with conventional tunnelling techniques, tunnels are normally set up to the end point, or destination node, to which a given data packet is addressed. However, this technique only works where all of the network nodes between the start of the tunnel and the destination node support at least one protocol in common. If not, tunnelled data packets will be discarded before the destination node is reached. Problems arise, therefore, when trying to apply conventional tunnelling techniques to a heterogeneous network, i.e. a network in which the network nodes do not necessarily support the same protocols and therefore do not necessarily have at least one protocol in common.

Ways of tunnelling IPv4 over IPv6 also exist, but they rely on using "IPv4 compatible" IPv6 addresses or on embedding the IPv4 address inside an IPv6 prefix. The present invention does not rely on these limitations.

It will be understood that the term "router" as used herein is intended to embrace a network element, or network node, (or part thereof) that is arranged to act as a data router.

It will further be understood that the term "protocol" as used herein is intended to embrace protocol set, or protocol stack, where the set (or stack) may comprise one or more protocols. For example, OSI and IP may each be considered to comprise a respective set of protocols, but may be referred to herein as OSI protocol or IP protocol respectively.

The invention relates particularly, but not exclusively, to OSI and IP protocols, and respective ISO and RFC standards may be obtained respectively from www.iso.ch and www.ietf.org, and the following terms are used hereinafter. An IP-only node is a node that can natively route IP packets but not OSI packets, particularly CLNP (ConnectionLess mode Network Protocol) packets, CLNP being an OSI network layer protocol. CLNP is the name given to the type of data packets or PDUs (Protocol Data Units) that are used to provide CLNS (ConnectionLess mode Network Service). CLNS is the service provided by the network layer of an OSI protocol stack to higher layers of the stack. Provision of CLNS service results in CLNP packets or PDUs being passed to lower layers of the stack. An OSI-only node is a node that can natively route OSI packets, particularly CLNP packets, but not IP packets. An IP capable node is a node that is able to route IP packets, and may or may not be able to route OSI packets too. An OSI capable node is a node that is able to route OSI packets, and may or may not be able to route IP packets too. A dual, or bi-lingual, node is a node that can natively route one or both of two protocols, particularly network layer protocols. This term is used

particularly to indicate either a node that routes both CLNS/CLNP and IPv4, or alternatively a node that routes both IPv4 and IPv6. A multi-lingual node is a node that can natively route one or multiples of two or more protocols, particularly network layer protocols, natively.

An adjacent network node is a reachable neighbouring node. The term "adjacency" is defined in section 3.6.3 of ISO/IEC 10589 and may be used herein to denote a reachable neighbouring node. A physical neighbouring node is not necessarily a valid adjacency, since it might be in a different network area or network level. Thus, an adjacency may be maintained between nodes that are not physical neighbours.

Summary of the invention

A first aspect of the invention provides an apparatus for routing data packets in a network comprising a plurality of nodes each arranged to support one or more of a plurality of sets of one or more protocols, the apparatus being included, in use, in a first network node that supports a first protocol set and one or more other protocol sets, the apparatus being arranged, upon receipt of a data packet conforming with the first protocol set at the first network node, to determine if the received data packet is destined for the first network node; the apparatus being further arranged to identify, upon determining that the received data packet is not destined for the first network node, a second network node, in a path to the destination node of the data packet, that

supports the first protocol set and at least one of the other protocol sets; whereupon the apparatus is arranged to cause the data packet to be encapsulated within a data packet conforming to the at least one other protocol set and having a destination address corresponding to the second node; and to cause the encapsulated data packet to be forwarded to the second network node.

Any other network node(s) between the first and the second network nodes are also arranged to support the at least one other protocol set in order to support the data tunnel created by said encapsulation.

Preferably, each protocol set comprises a network layer protocol. In this case, the apparatus is primarily concerned with identifying, as the second node, a node that supports the network layer protocol with which the original data packet conforms and a further network layer protocol that is also supported by the first node.

In the preferred embodiment, the protocol sets include OSI protocols, such as CLNS and CLNP, and/or IP protocols, such as IPv4 and/or IPv6.

It is also preferred that the respective paths to the other nodes are the 'shortest' paths as determined by a routing algorithm, for example the SPF (Shortest Path First) routing algorithm.

Preferably, the apparatus is arranged to select, as the second network node, the next, or nearest, suitable network node in the path to the destination node.

A second aspect of the invention provides a network node comprising an apparatus according to the first aspect of the invention.

In the preferred embodiment, the network node is arranged to support one or more Link State Protocols, such as IS-IS and/or Integrated IS-IS.

In an alternative embodiment, network nodes are arranged to distribute routing data packets carrying information identifying the data tunnelling capabilities of the respective network node from which they emanate. This arrangement is advantageously used with Link State Protocols including Integrated IS-IS, in which case said field may conveniently comprise a TLV-encoded variable length field in Integrated IS-IS LSPs, and OSPF (Open Shortest Path First), in which case said field is included in OSPF LSAs(Link State Advertisements). The tunnelling capability information may include whether or not the network node can terminate a data tunnel; what type of tunnelling protocols are supported by the network node; and which protocols may be encapsulated in said tunnels. Alternatively, said tunnelling capability information is conveyed indirectly by said data packets. For example, each network node that has data tunnelling capabilities is arranged to generate data packets, conveniently LSPs or LDAs, indicating that said network node has an adjacency

to one or more dummy network addresses depending on its data tunnelling capabilities, wherein one or more respective dummy addresses are used to signify respective tunnelling capabilities.

In a further alternative embodiment, each node in the network is arranged to indicate in its respective routing data packets an adjacency to one or more dummy network nodes depending on the tunnelling capability of the respective network node.

In the preferred embodiment, the network node may be arranged to act as a level-1 router and/or a level-2 router.

A third aspect of the invention provides a network comprising one or more network nodes according to the second aspect of the invention.

A fourth aspect of the invention provides a method of routing data packets in a network comprising a plurality of nodes each arranged to support one or more of a plurality of sets of one or more protocols, the method comprising: receiving, at a first network node that supports a first protocol set and one or more other protocol sets, the apparatus being arranged, a data packet conforming with the first protocol set; determining if the received data packet is destined for the first network node; identifying, upon determining that the received data packet is not destined for the first network node, a second network node, in a path to the destination node of

the data packet, that supports the first protocol set and at least one of the other protocol sets; encapsulating the data packet within a data packet conforming to the at least one other protocol set and having a destination address corresponding to the second node; and forwarding the encapsulated data packet to the second network node.

A fifth aspect of the invention provides a computer program product comprising computer useable instructions for causing a network node to perform the method of the fourth aspect of the invention.

A sixth aspect of the invention provides a computer program product comprising computer useable instructions for implementing, in whole or in part, the apparatus of the first aspect of the invention.

Preferred features of the invention are recited in the dependent claims.

In the preferred embodiment, the invention enables nodes that support routing of differing incompatible network layer protocols, such as CLNP, IPv4 or IPv6 to be present in a single routing domain, e.g. IS-IS level-1 area or level-2 subdomain, and provides nodes which automatically tunnel one network layer protocol over another as required, provided that all of the nodes in the routing domain support IS-IS and/or Integrated IS-IS routing protocols. This enables, in particular, IPv4-only, IPv6-only and OSI-only routers to be used in the same IS-IS level-1 area or level-2 subdomain, where Integrated IS-IS

is used to calculate routes across a network.

The invention does not affect the complexity of IP-only or OSI-only routers, but adds functionality to multi-lingual routers and in particular to dual, or bi-lingual, or tri-lingual routers.

With the present invention bi- or multi-lingual routers are arranged to automatically and dynamically forward data packets over a part of the network that does not support that packet type, to another bi- or multi-lingual router that is able recover the original data packet and forward into its final destination. Forwarding of data packets in accordance with the invention may be said to be dynamic in that a router sets up a tunnel to another router as and when required - the tunnels do not always exist. Further, a tunnel between two routers may be set up along a different path on successive occasions depending on the state of the network on each occasion.

Significantly, the bi- or multi-lingual router does not necessarily try to route the data packet to its destination node. Rather, the bi- or multi-lingual router is arranged to identify a second router (preferably the next or nearest), along a path to the destination node, that supports the protocol of the received data packet and also supports another protocol in common with the bi- or multi-lingual router itself. The bi- or multi-lingual router then tunnels the data packet to said identified second router using said other protocol.

By setting up a tunnel only as far as is necessary, the bi- or multi-lingual router is not required to know about the entire path to the destination node - the main concern is the path to the nearest network node that can unpack the tunnel. Thus, it does not matter if each node in the path to the destination node does not support a common protocol.

As will be appreciated from the following description of a preferred embodiment of the invention, a conventional routing algorithm for IS-IS routers, such as SPF (Shortest Path First), can readily be modified to identify said second node and to check if a tunnel can be set up and to cause a data packet to be encapsulated and forward on to the second node. The information that the algorithm needs to find this nearest node is conveniently already present in conventional Integrated IS-IS routing data packets such as the Hello and LSP packets.

The invention is particularly useful when mixing OSI and IP nodes or networks, or when migrating between one and the other. The invention is also particularly useful when migrating between IPv4 and IPv6. The invention also mitigates a perceived weakness in RFC 1195 that all routers in an IS-IS area must support all protocols present in the area. The invention enables data packets to be tunnelled over any nodes that are incompatible. It is possible that a data packet may go in and out of many data tunnels, or even tunnels within tunnels, as it travels to its destination.

The invention may be considered to provide auto-tunnelling in that a network node determines for itself a further (preferably the nearest or next) bi-lingual or multi-lingual node, in a path to a destination node, that is suitable to set up a tunnel with, and sets up a tunnel when required. The node achieves this by evaluating information received in routing data packets (and in particular LSPs) that are flooded across the network by each other network node.

Other aspects of the invention will become apparent to those ordinarily skilled in the art upon review of the following description of a specific embodiment and with reference to the accompanying drawings.

Brief description of the drawings

Figure 1 is a schematic diagram of a network, or routing domain, comprising a plurality of network nodes;

Figure 2 is a block diagram illustrating in part a network node and its operation;

Figures 3, 4 and 5 are block diagrams illustrating in part a network node and its operation as arranged in accordance with the present invention;

Figure 6 is a schematic diagram of four network areas linked by network nodes arranged in accordance with the invention; and

Figure 7 is a schematic view of two level-1 network areas including network nodes arranged in accordance with the invention.

Detailed description of the drawings

Referring firstly to Figure 1, there is shown a network 10, or routing domain, comprising a plurality of network nodes, or network elements (numbered 1 to 9 in Figure 1). The network nodes are arranged to route data packets (not shown) across the network 10. Each network node may comprise one or more piece of network equipment, such as a multiplexer or a cross-connect, but also comprises routing apparatus to enable it to route data packets across the network 10. Since the present invention relates particularly to the routing of data packets, the nodes (1-9), and in particular the routing apparatus included in the nodes, are referred to herein as routers. As will be appreciated from the following description, the routers may together form a level 1 IS-IS network area, or a level 2 subdomain, or may be separated into more than one level 1 network area. For illustrative purposes, the following description assumes that the network 10 is a level 1 network area.

The network 10 is also assumed to be heterogeneous in that each of the routers in the network 10 do not necessarily support a common protocol, particularly a network layer protocol such as CLNP (see ISO 8473-4), IPv4 (see RFC 791) or IPv6 (see RFC 2460).

The present example assumes that routers 1, 2, 3, 8 and 9 are standard IS-IS or Integrated IS-IS routers running protocol A, where A may be any of, for example, OSI, IPv4, IPv6 or any other protocol that is routable using IS-IS or Integrated IS-IS.

It is further assumed that routers 5 and 6 are standard IS-IS or Integrated IS-IS routers running protocol B, where B may be a different protocol of OSI, IPv4, IPv6 or any other protocol that is routable using IS-IS or Integrated IS-IS.

Routers 4 and 7 are assumed to be bi-lingual, or dual, routers that run Integrated IS-IS and can route both protocols A and B.

In accordance with the present invention, routers 4 and 7 have the following novel behaviour, which is described with reference to Figures 2 and 3 which show in part a network node and also illustrate its operation. In particular Figures 2 and 3 show those parts of the node that relate to the routing of data packets, i.e. a routing apparatus.

When router 4 receives a packet of, for example, type A (i.e. a data packet conforming with protocol A) from router 2 or 3 it determines whether the data packet is addressed to itself or whether the packet requires to be forwarded on (Fig.2, block 20). If router 4 determines that the data packet requires to be forwarded on, router 4 is arranged to inspect the destination address (i.e. the

address of the destination node which may be, say, router 8 or 9) of the packet and look up in its forwarding table (Forwarding Database - Fig. 2, blocks 24, 26, 28) which adjacent, or neighbouring, network node to send the packet to, in conventional manner.

As can be seen from Figure 2, a router includes a respective forwarding table 24, 26, 28, or database, in respect of each protocol that it supports. In the present example router 4 has two forwarding tables 24, 26 - one for each of protocols A and B. The third forwarding table 28 is required by nodes that support a third protocol, protocol C, where C may be a different protocol of OSI, IPv4, IPv6 or any other protocol that is routable using IS-IS or Integrated IS-IS.

Router 4 therefore needs to be able to determine which type (A,B or C) the data packet is and, to this end, includes a conventional protocol discriminator module 22 for distinguishing between protocols. It will be noted that the data packet at this point may or may not comprise one protocol encapsulated within one or more other protocols - the protocol discriminator 22 needs to determine which protocol is the 'outermost' encapsulating protocol.

Once the protocol of the data packet has been established (which in the present example is protocol A), the router 4 refers to the forwarding table that is appropriate to the established protocol (table 24 in the present example).

Before sending the packet onto its next hop (i.e. the selected adjacent or neighbouring router), router 4 is arranged to check that the next hop can route, or deal with, the packet. Conveniently, it will do this by inspecting "Hello" packets from that neighbouring router, or adjacency, and in particular by inspecting the "protocols supported" field in the Hello packet, to establish if the next hop supports protocol type A (Fig. 3, block 30). To this end, the router 4 includes a table, database or other storage means (Fig.3, block 32), for storing a list of protocols that its neighbour nodes, or adjacencies, support - the table 32 being created and updated using information contained in the "Hello" packets received from its neighbours, or adjacencies. If a "Hello" packet does not contain a "protocols supported" field then, in the preferred embodiment, the router assumes that the neighbouring node in question is an OSI-only router. "Hello" packets are standard routing data packets that are distributed by each node, or router, to its neighbouring nodes in accordance with the conventional IS-IS routing protocols. Alternatively, the router can gather the information concerning supported protocols by examining the information carried in other routing data packets that are distributed across the network by each node. For example, the Link State PDUs (LSPs) that are distributed by nodes in accordance with Integrated IS-IS include a "protocols supported" field and may be used for this purpose.

If the next hop does support protocol A, then the router 4 will forward the data packet to the next hop in

conventional manner and without the need for data tunnelling (Fig.3, block 33).

If the next hop does not support protocol A, but supports protocol B (Fig. 3, block 34), then router 4 is arranged to encapsulate the original packet of type A inside a new packet of type B, using an encapsulation technique such as, but not limited to, GRE (Generic Routing Encapsulation - see RFC 1701 (updated to RFC 2784), RFC 1702 and RFC 3147) (Fig. 3, block 35). The new encapsulating packet conforms to protocol B in this example (typically a network layer protocol) and comprises a destination address and a source address to encapsulate the original data packet. The network layer protocol of the new packet is one that is supported by the next hop as defined by the "protocols supported" TLV of Hello (or other) PDUs received from the next hop. The destination address of the new packet corresponds to the identity of the next node along the shortest path to the original destination that supports both the type of network layer protocol that the original packet conforms with, and a network layer protocol that is supported by the next hop (as defined by the "protocols supported" TLV of Hello PDUs received from the next hop) (Fig. 3, block 42). Preferably, this is achieved by inspection of the "protocols supported" TLV of LSPs received from all of the nodes in the path to the destination, and identifying the first that meets the above requirement. The source address of the new packet corresponds with the identity of the node that constructs the new encapsulation packet.

It is noted that the encapsulation of one packet inside another may result in a new packet that is longer than the MTU (Maximum Transmission Unit) size of the link over which the new packet must be forwarded. Since the new encapsulating packet (encapsulated using RFC 1702 or RFC 3147, for example) packet should not be discarded, the packets should not have the "Don't Fragment" bit set if they are IP packets and should have the "Segmentation Permitted" flag set if they are CLNP packets. The resultant encapsulated packets should then be fragmented before being forwarded if the packet is now longer than the MTU limit of the link.

In the case where router 4 also supports protocol C, the router 4 may proceed to block 36 to check whether or not the adjacency supports protocol C (this may occur either if router 4 does not support B or if it is determined that the adjacency does not support B). If the adjacency does not support A, B or C, then an error is signalled (Block 37).

In the present example, should router 4 determine that the next hop, or adjacency (e.g. node 6), in the path to the destination node supports protocol B (and not A), router 4 identifies (in accordance with the preferred embodiment) the first, or nearest or next, network node in said path that supports both protocols A and B (say router 7 in the present example). The router 4 is able to tunnel the type A data packet over the incompatible adjacency (router 6) by encapsulating the type A data packet within protocol B and sending it to router 7. This is possible because

router 7 is bi-lingual and supports the protocol of the data packet (A) and has an additional commonly supported protocol with router 4, namely protocol B. Router 7 is therefore able to unpack the tunnel. It will be noted that the incompatible adjacency (router 6) must also support said additional commonly supported protocol (B in this case) so that it can carry the tunnel.

Router 4 therefore needs to be able to identify the next appropriate bi-lingual router in the path to the destination node. This is achieved by modifying the routing algorithm run by the router 4 as described below in the context of the SPF (Shortest Path First) routing algorithm, which is described in standards ISO 10589 and RFC 1195.

When router 4 runs the SPF algorithm typically used by IS-IS, it will perform some extra novel functions in accordance with the invention.

When a network node, or router, runs SPF, it examines every possible path across the network to each other network node in order to determine the most appropriate, or 'shortest', path to that node. It does this by evaluating the information contained in Link State PDAs (LSPs) that are flooded across the network by each network node and gathered by each other network node in accordance with IS-IS and integrated IS-IS. When the shortest path has been identified, the router stores, in a database commonly known as PATHS, identification of to which neighbouring node, or adjacency, a data packet should be

sent in order to reach a given destination node by the 'shortest' path. Typically, there is an entry in the PATHS database for each network node, and each entry comprises an identifier for the node, an identifier for the appropriate adjacency and an identifier for cost (metric).

In accordance with the invention, for each and every path that router 4 calculates, or evaluates, it is arranged to look for another router along that path that supports both protocols A and B. It can do this by inspecting the respective LSPs that it has received from each router along the path. If an LSP includes a "protocols supported" field then that will state which protocols the router that sent the LSP supports. If an LSP does not include a "protocols supported" field then the router that sent the LSP is an OSI-only router. In the preferred embodiment, the router looks for the first (i.e. next or nearest) suitable router in the path.

For each 'shortest' path (as determined by the SPF algorithm in normal fashion) router 4 is arranged to store the identity of the first router along that path that supports both protocols A and B. A convenient place to store this information is in the PATHS database as specified in RFC 1195 and ISO 10589 - for example by adding an extra field to the entry for the node (router) in question, which field includes an identifier for said first, or next, bi- or multi-lingual router.

Router 4 may store the identity of said next bi- or multi-lingual router in the form of destination addresses expressed in the formats of both protocols A and B, or it may store it in only one format, or some other format and use another translation table (not shown) when needed in order to obtain a destination address in format A or B. A preferred solution is to add as many extra columns, or fields, as necessary to the PATHS database to identify the first router that supports A and B, the first that supports A and C, the first that supports B and C, and to express them as identifiers of the form used in Integrated IS-IS SPF calculations. This implies a number of extra columns dependant upon the number of protocols supported by the router. For example, if the first router along the path that supports A and B also supports C, then the identity of that router will appear in A&B, A&C and B&C columns in the PATHS database. In the case of router 4, which only supports A and B, then the PATHS database only requires an additional field to identify the next router that supports A and B and does not require an entry in a field for A and C, since router 4 does not support C. A router may conveniently be arranged to ignore any information in a received LSP that relates to a protocol that it does not support.

In the preferred embodiment, each bi- or multi-lingual router compiles and stores information (conveniently in the PATHS database) in respect of the 'shortest' path to each other node in the network, which information identifies the next bi- or multi-lingual router in the respective shortest path that supports at least two

protocols in common with itself. Where a router supports more than two protocols, it stores said identifying information in respect of each protocol that it supports paired with each other protocol that it supports. For example, if a router supports protocols A, B and C, it stores information identifying the next node that supports A and B, the next node that supports A and C, and the next node that supports B and C, in respect of each shortest path. Clearly, depending on the structure of the network, there may not always be one of each type in each path.

It will be noted that the router, or node, does not necessarily need to forward an encapsulated data packet to the first, or next, suitable node in the path to the destination node. For example, if the next suitable node is congested, the router may select to forward the encapsulated data packet to a further suitable network node in the path to the destination node. Thus, the router is preferably arranged to store, conveniently in PATHS, information identifying one or more other nodes in the path to a destination node, which other nodes support at least two protocols in common with itself.

The LSP, the modified SPF algorithm and the modified PATHS database are represented in Figure 3 by blocks 38, 39 and 40 respectively. It will be understood that the information identifying the next suitable bi- or multi-lingual router need not necessarily be stored in the PATHS database but may alternatively be stored in any other convenient storage means. It will further be understood that a router may contain a separate forwarding table (not

shown), or equivalent storage means, for each individual protocol that it supports. This would result in tunnel interfaces automatically appearing in the forwarding table for one particular protocol for certain destinations, whilst different or possibly no tunnel interfaces may appear in the forwarding table for another protocol for entirely different destinations.

Thus, referring again to Figure 3, and in particular to block 41, router 4 is arranged to inspect the modified, or extended, PATHS database 40 to determine the identity of the next router along the path that supports both protocols A and B. When so determined, router 4 uses the address of said next router as the destination address of the new packet of type B (Fig. 3, block 42) before it sends the encapsulated data packet to the next hop (Fig. 3, block 33). As mentioned above, the router may have to refer to another table to translate the identity of the router expressed in the first table (PATHS database) into a format that represents a destination address suitable for a packet of protocol type B.

Referring now to Figure 3, block 43, in a preferred embodiment, the router may be arranged to check whether or not there is in fact a next node in the path that supports A and B and, if not, to check if the adjacency supports protocol C (Fig. 3, block 36). This is only a relevant option for router 4 if it were to also support protocol C.

It will be appreciated that blocks 45, 46, 47 and 48 of Figure 3 correspond with blocks 43, 35, 42 and 41 but in respect of protocols A and C rather than A and B.

Similarly, Figures 4 and 5 correspond substantially with Figure 3 but relates to the cases where the received data packet is type B and type C respectively.

Router 7, or any other router in the IS-IS area (or level-2 subdomain) that supports more than one protocol will have similar functionality for sending packets in the reverse direction, or onwards in similar manner.

In the present example router 4 will identify router 7 as being the next router, along its paths for either of routers 8 or 9, that supports both protocols A and B. Therefore, in the PATHS database associated with router 4, the respective entry for router 8 and for router 9 will each include a field in which router 7 is identified as the next suitable bi-lingual router.

Router 4 will send any packets of type A destined for either routers 8 or 9 encapsulated inside a new packet of type B with destination address set to the identity of router 7.

Router 7 is arranged to receive these encapsulation packets and is therefore required to have the following novel behaviour.

Router 7 is arranged to examine or inspect any incoming packet that has its destination address set to itself (it will be noted that this destination address is as set by router 4 at Fig. 3, block 42 and is not necessarily the same as the address of the ultimate destination node of the original data packet) - this is determined in Fig.2, block 20. If the data packet is addressed to router 7, router 7 is further arranged to inspect the data packet to see if it contains another packet encapsulated inside it (Fig.2, block 21. If it does then router 7 de-encapsulates the packet inside (Fig.2, block 23) in conventional manner and reprocesses the de-encapsulated packet as if it had just been received by router 7 (return to block 20, Fig.2). If not then router 7 treats the data packet in conventional manner as an exception packet (Fig. 2, block 25).

In this way data packets destined for routers 8 or 9, but encapsulated inside packets of type B will be de-encapsulated by router 7, and will then be processed and forwarded as normal, as packets of type A.

Router 4, or any other router in the IS-IS area (or level-2 subdomain) that supports more than one protocol will have similar functionality for receiving packets in the reverse direction.

The preferred embodiment of the invention, as illustrated in and described with reference to Figures 2-5, may be described as a routing apparatus and may conveniently be implemented by modification of, in particular, the SPF

algorithm and the PATHS database. The following description provides an example of how this may be achieved. As part of the SPF algorithm each node identifies one or more paths for every destination node, and stores respective path entries into a table known as the TENT table, or database. For each destination node, the or each path entry in TENT is then compared with the entry in PATHS. The entry corresponding to the shortest of the two paths is put into PATHS and the longest deleted. When TENT is empty then the SPF algorithm is finished. At this stage every destination node has a respective entry in PATHS that describes the complete shortest path from the node to the destination node. It is noted that PATHS may have more than one "shortest" path entry for a destination node if two or more paths are found to have the same cost.

Normally the first SID (System Identifier) in each path after the node is loaded into the forwarding table for each destination node (IP forwarding table if a path to an IP address, OSI forwarding table if a path to an OSI End System), as it is the next hop.

To implement the invention, the algorithm may be modified to jump along all of the nodes in each final entry in PATHS, and for each path retrieve the LSP for each node along the path, and examine if the node supports both the network layer protocol of the type that is the destination address, and a network layer protocol of the type that is one supported by the next hop (as specified in the "protocols supported" TLV present in IS-IS Hello PDUs (or LSPs) received from that node). The first node along the

path that it finds that fulfils this requirement is selected as the next dual node along the shortest path to the final destination. Once the first node along the path is found that fulfils the requirements, the process may stop for that path.

This additional process only need happen when the next hop does not support the network layer protocol of the type that corresponds to the destination address for that path. If the next hop does support that type of network layer protocol (as specified in the "protocols supported" TLV present in IS-IS Hello PDUs (or LSPs) received from that node) then packets to that destination may simply be forwarded natively and forgotten, and so the search for a next dual node along the path is not necessary.

The algorithm is arranged then to identify an IP address for the next dual node if the destination of the path is an OSI End System, and must then identify an OSI address for this next dual node if the destination of the PATH is an IP address.

For each IP destination that requires encapsulation to get beyond the next hop, the node can then put a marker in the IP forwarding table indicating the OSI destination address that must be used to encapsulate all IP packets destined for that address.

For each OSI destination that requires encapsulation to get beyond the next hop, the node can then put a marker in the OSI forwarding table indicating the IP destination address that must be used to encapsulate all OSI packets destined for that address.

A node that supports IPv4, IPv6 and OSI may find two addresses (for example an IPv4 address and an IPv6 address) that could be used for encapsulation. In this case it may choose either as long as it results in a packet that is of a network layer protocol type that the next hop supports (as specified in the "protocols supported" TLV present in IS-IS Hello PDUs (or LSPs) received from that node).

In the preferred embodiment, the network 10 and the network nodes exhibit the following features to comply with IS-IS and Integrated IS-IS.

OSI-only nodes are conformant with ISO/IEC 10589 and IP-only nodes are conformant with RFC 1195.

IS-IS or Integrated IS-IS routers that are running level-1 routing cannot see LSPs from other routers outside of their own IS-IS area. Therefore a router that acts as a level-1 and a level-2 router should support all of the network layer protocols that are present in its IS-IS area. This applies to routers on both sides of any such boundary. As, in accordance with the preferred embodiment, a bi- or multi-lingual router finds the first router along a path that supports the protocol of the data packet that requires forwarding and the protocol that provides the dynamic tunnel, and sends all packets that require tunnelling to that router; it is then possible to guarantee that no dynamic tunnel arising from this mechanism will extend beyond a router that supports level-1 and level-2 routing. Therefore there will be no packets

encapsulating other packets traversing the boundary between level-1 area and level-2 subdomain. Therefore it does not matter that level-1 LSPs do not traverse the area boundaries. Also, it does not matter that level-2 LSPs do not travel outside of the level-2 subdomain.

Also, all routers in the IS-IS area that run Integrated IS-IS and support more than one protocol (and in particular more than one network layer protocol) are required to implement the invention, as described above in the exemplary cases of routers 4 and 7.

In a level-2 subdomain, either all routers must support all network layer protocols present, or all routers that support more than one protocol (and in particular more than one network layer protocol) are required to implement the invention.

To avoid loss of data packets, routers should not be directly connected to other routers that do not support at least one protocol in common, particularly a network layer protocol. This means, for example, that an OSI-only router should not be directly connected directly to an IP-only router. If a data path is wanted between an OSI-only router and an IP-only router then a bi- or multi- lingual IP and OSI router should be installed between them. A LAN (Local Area Network) is a direct connection. If a LAN does have both IP-only and OSI-only nodes present, the IP-only nodes should not form an adjacency with the OSI-only nodes. This results in two separate pseudonode election processes on the LAN. Any bi-lingual (dual node) (i.e.

OSI and IP) router on the LAN must be able to take part in both of the pseudonode election processes. One method of ensuring that, say, OSI-only routers and IP only routers are not connected directly together is to arrange that they each operate different layer 2 protocols - for example LAPD (link access procedure D) for the OSI only routers and PPP (point-to-point protocol) for the IP only routers. In this case, bi- or multi-lingual routers may be arranged to run both LAPD and PPP. Alternatively, each Integrated IS-IS node is required to check if its neighbouring node(s) support the network layer protocol that it supports itself. If there is no common network layer protocol with a neighbouring node, then the node refuses the adjacency.

In the preferred embodiment, nodes that implement the invention are arranged to examine the "protocols supported" field in the "Hello" PDUs (or LSPs) received from neighbouring nodes, and if none of the network layer protocols that the node itself supports are listed, then the node refuses the adjacency.

Nodes that are conformant to RFC 1195 but which do not implement the invention may be used in mixed level-1 areas or level-2 subdomains with the following restrictions. Integrated IS-IS nodes that support only one network layer protocol but which do not conform with the invention may still be used in an level-1 area or level-2 subdomain, but it should be ensured that such a node does not have any adjacencies with other nodes that might forward packets to it that it does not support. This may be achieved

manually by a network manager (not shown), for example, or may be achieved automatically by the network nodes themselves - for example, some RFC 1195 compliant routers are arranged to refuse unsuitable adjacencies.

Integrated IS-IS nodes (or clusters of nodes) that support more than one network layer protocol but which do not conform with the invention may still be used in an level-1 area or level-2 subdomain when they have adjacencies only with other nodes that support only one of those network layer protocols, if all of those adjacent nodes support the same network layer protocol in common. Integrated IS-IS nodes (or clusters of nodes) that support more than one network layer protocol but which do not conform with the invention may still be used in an level-1 area or level-2 subdomain when they are surrounded by other dual nodes on all adjacencies that are conformant with the invention.

It is possible for a router to support three protocols, A, B and C.

Such a router must be able to terminate tunnels of C over A and C over B, even though it may not be directly connected to another router supporting protocol C. The router may find itself in a position in the network where it is connected to other routers that support, say, type A in one direction and routers that support type B in another direction. In this case, the router is arranged to advertise to the other routers of type A that it supports A, B and C, as well as advertising into routers of type B that it supports A, B and C. In this example

the router is required to be able to terminate a dynamic tunnel containing type C packets encapsulated within type A and route the type C packet onwards, which may involve it putting the type C packet back into an encapsulation, but this time inside a packet of type B.

This is illustrated by way of example in Figure 6.

Network 1 is a network of routers that support protocol C only. Network 2 is a network of routers that support protocol A only. Network 3 is a network of routers that support protocol B only. Network 4 is a network of routers that support protocol C only. Routers 51, 52, 53 are bi- or multi-lingual routers on the boundaries between networks 1, 2, 3 and 4, as shown.

If router 51 supports protocols A and C, router 52 supports protocols A, B and C, and router 53 supports B and C, then router 51 receives LSPs from router 52 and recognises router 52 as capable of routing protocol A and C. Router 53 receives LSPs from router 52 and recognises router 52 as capable of routing protocols A, B and C.

When a node in network 1 wishes to send a packet of protocol type C to a node in network 4 it is forced by the layout of the network to send the packet via router 51. Router 51 looks for the next router along the shortest path that supports A and C. Thus it finds router 52. Router 51 encapsulates the packet of protocol type C inside a packet of protocol type A and sends it to router 52.

Router 52 receives the packet of protocol type A and looks inside it and finds another packet of protocol type C that is destined for somewhere else.

Despite not having any type C direct connections, router 52 is capable of routing packets of protocol type C and so does so.

Router 52 looks for the next router along the shortest path that supports B and C. It finds router 53. Router 52 encapsulates the packet of protocol type C inside a packet of protocol type B and sends it to router 53. Router 53 receives the packet of protocol type B and looks inside it and finds another packet of protocol type C that is destined for somewhere else.

Router 53 then forwards the original packet of protocol type C into network 4 which supports protocol C and it reaches its destination node in the normal way.

An alternative scenario is that router 51 supports A and C, router 52 supports A and B and that router 53 is the one that supports A, B and C.

When a node (or router) in network 1 wishes to send a packet of protocol type C to a node in network 4 it is forced by the layout of the network to send the packet via router 51. Router 51 looks for the next router along the shortest path that supports A and C. It finds router 53. Router 51 encapsulates the packet of protocol type C

inside a packet of protocol type A and sends it to router 53.

Due to the constraint in the network layout the packet of type A that has the packet of type C inside it arrives at router 52.

The destination address of the packet has been set to 53, and not 52, so router 52 does not look inside it but routes it onwards as it would any other packet of type A. Router 52 looks for the next router along the shortest path that supports A and B. It finds router 53. Router 52 encapsulates the packet of protocol type A inside a packet of protocol type B and sends it to router 53. Router 53 receives the packet of protocol type B and looks inside it and finds another packet of protocol type A that is also destined for it.

Router 53 looks inside the packet of protocol type A and finds another packet of protocol type C that is destined for somewhere else.

Router 53 then forwards the original packet of protocol type C into network 4 which supports protocol C and it reaches its destination node in the normal way.

With reference now to Figure 7, it is described, by way of example, how traffic (data packets) may cross boundaries between IS-IS areas in a heterogeneous network.

Routers 81, 82, 83 and 84 are assumed to support only protocol A.

Routers 71, 72, 73, 74, 91 and 92 are assumed to support only protocol B.

Routers 61, 62, 63, 64, 65 and 66 are assumed to support both protocols A and B.

Routers 62, 63, 64 and 65 are also arranged to act as level-1 and level-2 routers.

Routers 91 and 92 are level-2 only routers.

Routers 62, 63, 64 and 65 are required to run level-1 and level-2 routing, as is normal IS-IS practice, in order to link IS-IS area 1 and IS-IS area 2 together across the level-2 subdomain that is formed by routers 62, 63, 64, 65, 91 and 92.

In accordance with the requirements mentioned above, routers 62, 63, 64 and 65 are also required to support all of the protocols present in their IS-IS area because they are level-1 and level-2 routers. Therefore routers 62, 63, 64 and 65 must support both protocols A and B.

If router 81 sends a protocol A packet to router 83, then the packet has to pass through router 61.

Router 61 determines that neither routers 71 or 72 support protocol A and so is forced to send the packet onwards into the network encapsulated inside a packet of type B.

For this router 61 searches along the shortest path until it finds a router that supports both protocols A and B. This is guaranteed to be either router 62 or router 63. For the purposes of this example it is assumed that it 62 is on the shortest path.

Router 62 receives the encapsulated packet and removes the encapsulation thus recovering the original packet of type A.

In accordance with the requirements set out above, all level-1 and level-2 routers must be able to route all protocols that are present in their IS-IS area. Therefore both routers 61 and 62 must support both protocols A and B.

This guarantees that any encapsulation is removed when the packet arrives at router 61 or 62 as is mentioned above.

Router 62 finds the shortest path to a level-2 router that is in IS-IS area 2. This is to router 64 via router 91. Router 91 does not support protocol A, so router 62 looks for the next router along the path that does. Router 62 sends the packet to router 64 encapsulated within a packet of type B which can be routed by router 91.

Router 64 receives and de-encapsulates the original type A packet. Router 64 forwards the packet onwards into its area towards router 83. It also encapsulates the packet to get it through router 73 which does not support protocol A.

It will be noted that when a router acts as a level-1 router, it is arranged to generate and distribute level-1 type LSPs. While a router acting as a level-2 router is arranged to generate and distribute level-2 LSPs.

In the embodiment described above, the routers use the "Protocols Supported" field of LSPs generated in accordance with Integrated IS-IS (RFC 1195) in order to determine which protocol(s) are supported by the other routers in the network. This is convenient since the "Protocols Supported" field is normally included in Integrated IS-IS LSPs and so no modification to standard LSPs is required. However, for this approach to work, an assumption is made that if a router advertises via the "Protocol Supported" field that can support, for example, protocols A and B, then it can also create, terminate, or otherwise process, data tunnels comprising protocol A encapsulated within protocol B, and vice versa. That is, it is assumed that the router comprises the necessary components (such as GRE encapsulation and decapsulation software) to enable it to process data tunnels.

It is possible, however, for routers to use means other than the "Protocols Supported" field to determine the tunnelling capability of other routers. One option is to create, or define, a new field in the LSP particularly for

carrying information concerning the tunnelling capabilities of the router that issues the LSP. In this way a router can use the new LSP field to advertise to other routers that, for example, it can terminate tunnels, what type of tunnelling protocols it supports, and protocols that may be encapsulated in said tunnels.

The addition of a new "tunnelling capability" field is conveniently achievable in Integrated IS-IS by defining a new TLV-encoded (Type Length Value - encoded) variable length field.

Moreover, the provision of a "tunnelling capability" field is not limited to use with Integrated IS-IS. Other Link State Protocols, such as Open Shortest Path First (OSPF), also use Link State PDUs (known as Link Sate Advertisements (LSAs) in the case of OSPF) to enable communication amongst routers. The invention is therefore particularly suited for use with Link State Protocols wherein it is possible to modify an LSP, or equivalent data packet, to include a field for carrying tunnelling capability information.

It will be appreciated from the foregoing that encapsulation and tunnelling takes place only when deemed necessary by a router, and thus the tunnels are automatically created and are dynamic. The tunnels do not need to be maintained and may exist only when a data packet requires one. In the preferred embodiment, as packets still cross the network along the shortest path that each node calculates in normal manner, and as all of

the nodes have a basic routing protocol in common, there is no need for IS-IS packets to traverse these tunnels, only IP and CLNS/CLNP traffic is encapsulated.

It will be understood therefore that the present invention is not limited to use with Integrated IS-IS. For example, the invention may be used to tunnel IPv6 over IPv4 (or vice versa) using OSPF.

A further alternative is to arrange that a router advertises to other routers its tunnelling capabilities by means of advertising adjacencies to one or more dummy address. For example, a router that is capable of terminating a tunnel of a particular type may advertise this fact by indicating in the LSPs that it generates that it has an adjacency to a pre-determined dummy address. Thus, a router that required to initiate a tunnel of said particular type would search its LSP database for another router that has advertised an adjacency to said pre-determined dummy address. This method has the advantage that the standard LSPs do not need to be modified. It is sufficient that each router that sets up tunnels can recognise dummy addresses and associate them with corresponding tunnelling capabilities. This approach is also particularly suitable for use with Link State Protocols, such as Integrated IS-IS or OSPF.

The invention is not limited to the embodiments described herein which may be modified or varied without departing from the scope of the invention.

The following standards are relevant to the foregoing description and are hereby incorporated herein by reference:

ISO/IEC 10589, which describes how SPF can be applied to routing OSI and specifies the resulting routing protocol that is IS-IS, and also describes the PATHS database and IS-IS LSPs;

RFC 1195, which describes how standard ISO 10589 can be extended to route IP (in particular IPv4) as well as OSI, and further describes the PATHS database and the extensions to the IS-IS LSPs that are needed to specify the routing protocol that is Integrated IS-IS (also known as Dual IS-IS);

RFC 1701 and the updated version RFC 2784, which describe GRE; RFC 1702 and RFC 3147 which relate to the routing of GRE encapsulated data packets; and

http://www.ieft.org/internet-drafts/draft-ieft-isis-ipv6-01.txt and http://www.ietf.org/internet-drafts/draft-ietf-isis-ipv6-02.txt, which are draft documents discussing how Integrated IS-IS may be modified to support IPv6.

CLAIMS:

1. An apparatus for routing data packets in a network comprising a plurality of nodes each arranged to support one or more of a plurality of sets of one or more protocols, the apparatus being included, in use, in a first network node that supports a first protocol set and one or more other protocol sets, the apparatus being arranged, upon receipt of a data packet conforming with said first protocol set at said first network node, to determine if said received data packet is destined for said first network node; the apparatus being further arranged to identify, upon determining that said received data packet is not destined for said first network node, a second network node, in a path to the destination node of said data packet, that supports said first protocol set and at least one of said other protocol sets; whereupon the apparatus is arranged to cause said data packet to be encapsulated within a data packet conforming to said at least one other protocol set and having a destination address corresponding to said second node; and to cause the encapsulated data packet to be forwarded to said second network node.

2. An apparatus as claimed in Claim 1, wherein the apparatus is arranged to select said at least one other protocol set by determining which protocol sets are supported by an adjacent network node in the path to the destination node.

3. An apparatus as claimed in Claim 2, wherein the apparatus is arranged to identify a second protocol set supported by an adjacent network node, said second protocol set being included in said one or more other protocol sets supported by said first network node; the apparatus being further arranged to select, as said second network node, a network node in the path to the destination node that supports said first and second protocol sets.

4. An apparatus as claimed in Claim 1, whereupon determining that said received data packet is destined for said first network node, the apparatus is arranged to determine if said received data packet contains data conforming with another protocol set encapsulated within said first protocol set and, upon so determining, to cause said received data packet to undergo de-encapsulation; the apparatus being further arranged to examine whether or not the de-encapsulated data packet is destined for said first network node.

5. An apparatus as claimed in any preceding claim, wherein each protocol set comprises a network layer protocol.

6. An apparatus as claimed in any preceding claim, wherein said protocol sets include OSI protocols and/or IP protocols.

7. An apparatus as claimed in Claim 6, wherein said OSI protocols include CLNS (ConnectionLess mode Network

Service) and CLNP (ConnectionLess mode Network Protocol) and said IP protocols include IPv4 and/or IPv6.

8. An apparatus as claimed in any preceding claim, further arranged to compile information in respect of each adjacent node in the network, which information includes identification of the or each protocol set supported by the respective adjacent node.

9. An apparatus as claimed in Claim 8, wherein said information is compiled by examining information contained in respective routing data packets that are distributed, in use, by each adjacent network node.

10. An apparatus as claimed in any preceding claim, being further arranged to compile, in respect of each other node in the network, information identifying one or more network nodes in a path to said other node that support at least two protocol sets in common with itself, said second network node being selected from said one or more network nodes.

11. An apparatus as claimed in Claim 10, wherein the respective paths to said other nodes are the 'shortest' paths as determined by a routing algorithm.

12. An apparatus as claimed in Claim 10 or 11, wherein the apparatus is arranged to select, as said second network node, the next, or nearest, suitable network node in the path to the destination node.

13. An apparatus as claimed in any of Claims 10 to 12, wherein the apparatus is arranged to compile said information by evaluating information contained in routing data packets that are distributed, in use, by each network node.

14. A network node comprising an apparatus as claimed in any of Claims 1 to 13.

15. A network node as claimed in Claim 14, wherein the network node is arranged to support one or more Link State Protocols and said routing data packets comprise Link State packets.

16. A network node as claimed in Claim 15, wherein the network node is arranged to support IS-IS protocol and the Link State packets comprise IS-IS 'Hello' packets.

17. A network node as claimed in Claim 16, wherein the network node is arranged to support IS-IS protocol and/or Integrated IS-IS protocol and the Link State packets comprise Link State PDUs (Protocol Data Unit) (LSP).

18. A network node as claimed in Claim 17, wherein said information is compiled from the "protocols supported" field of Integrated IS-IS LSPs.

19. A network node as claimed in any one of claims 14 to 18, wherein said routing data packets are arranged to carry information identifying the data tunnelling

capabilities of the respective network node from which they emanate.

20. A network node as claimed in Claim 19, wherein each node in the network is arranged to indicate in its respective routing data packets an adjacency to one or more dummy network nodes depending on the tunnelling capability of the respective network node.

21. A network node as claimed in any one of claims 14 to 20, further comprising means for storing said compiled information in respect of each other node in the network.

22. A network node as claimed in Claim 21, wherein said storage means comprises the PATHS database provided for by IS-IS and Integrated IS-IS.

23. A network node as claimed in Claim 21 or 22, wherein the routing apparatus is arranged to store, in said storage means and in respect of each other node in the network, a respective identifier of one or more suitable second network nodes.

24. A network node as claimed in Claim 23, wherein the routing apparatus is arranged to store a respective identifier of a suitable second network node in respect of each pair of protocol sets supported in common by the network node and the respective other network node.

25. A network node as claimed in Claim 23 or 24, wherein the network node is arranged to store identifiers of

suitable second network nodes that are next, or nearest, in the respective path to the other network node.

26. A network node as claimed in any of Claims 14 to 25, wherein the network node is arranged to act as a level-1 router.

27. A network node as claimed in any of Claims 14 to 26, wherein the network node is arranged to act as a level-2 router.

28. A network comprising one or more network nodes as claimed in any of Claims 14 to 27.

29. A method of routing data packets in a network comprising a plurality of nodes each arranged to support one or more of a plurality of sets of one or more protocols, the method comprising:

receiving, at a first network node that supports a first protocol set and one or more other protocol sets, the apparatus being arranged, a data packet conforming with said first protocol set;

determining if said received data packet is destined for said first network node;

identifying, upon determining that said received data packet is not destined for said first network node, a second network node, in a path to the destination node of

said data packet, that supports said first protocol set and at least one of said other protocol sets;

encapsulating said data packet within a data packet conforming to said at least one other protocol set and having a destination address corresponding to said second node; and

forwarding said encapsulated data packet to said second network node.

30. A computer program product comprising computer useable instructions for causing a network node to perform the method of Claim 29.

31. A computer program product comprising computer useable instructions for implementing, in whole or in part, an apparatus as claimed in any of Claims 1 to 13.

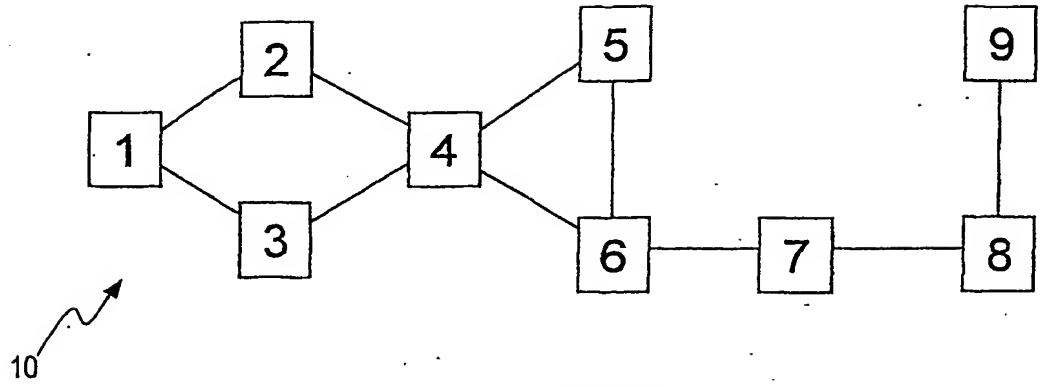

1
2
3
4
5
6
7
8
9
10

Fig.1

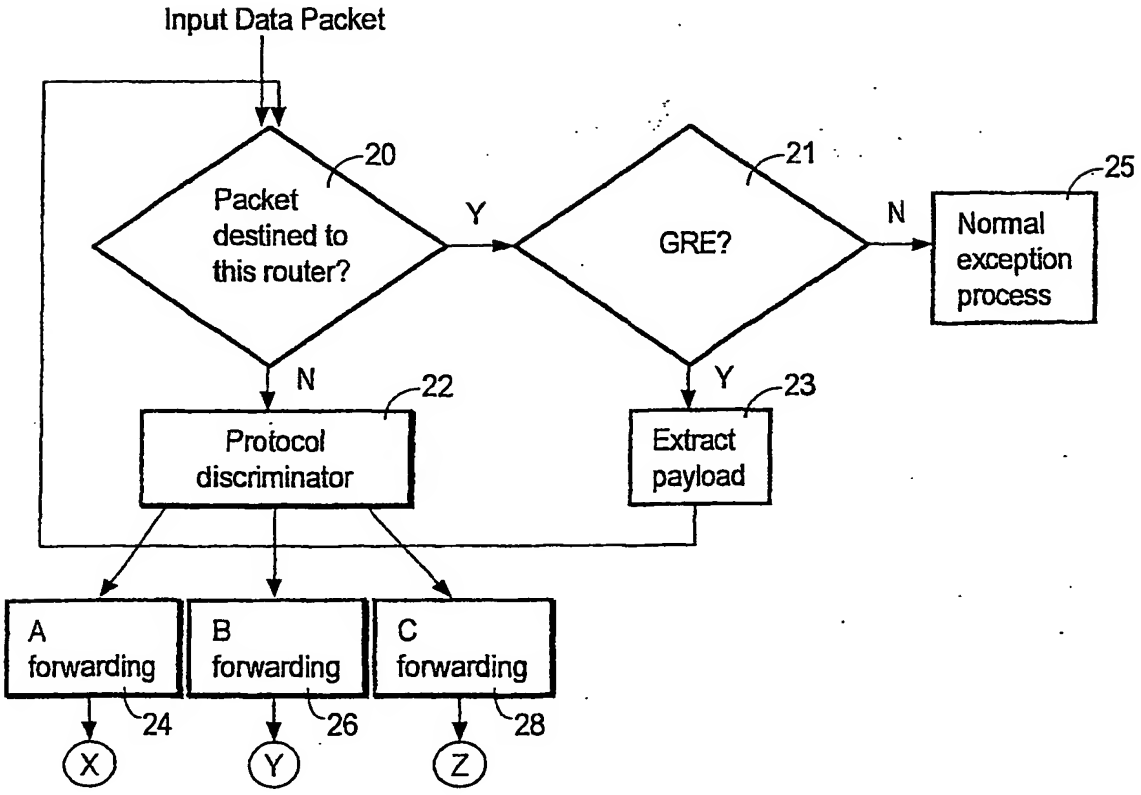

Input Data Packet
Packet destined to this router?
20
Y
GRE?
21
N
Normal exception process
25
N
22
Protocol discriminator
Y
23
Extract payload
A forwarding
B forwarding
C forwarding
24
26
28
X
Y
Z

Fig.2

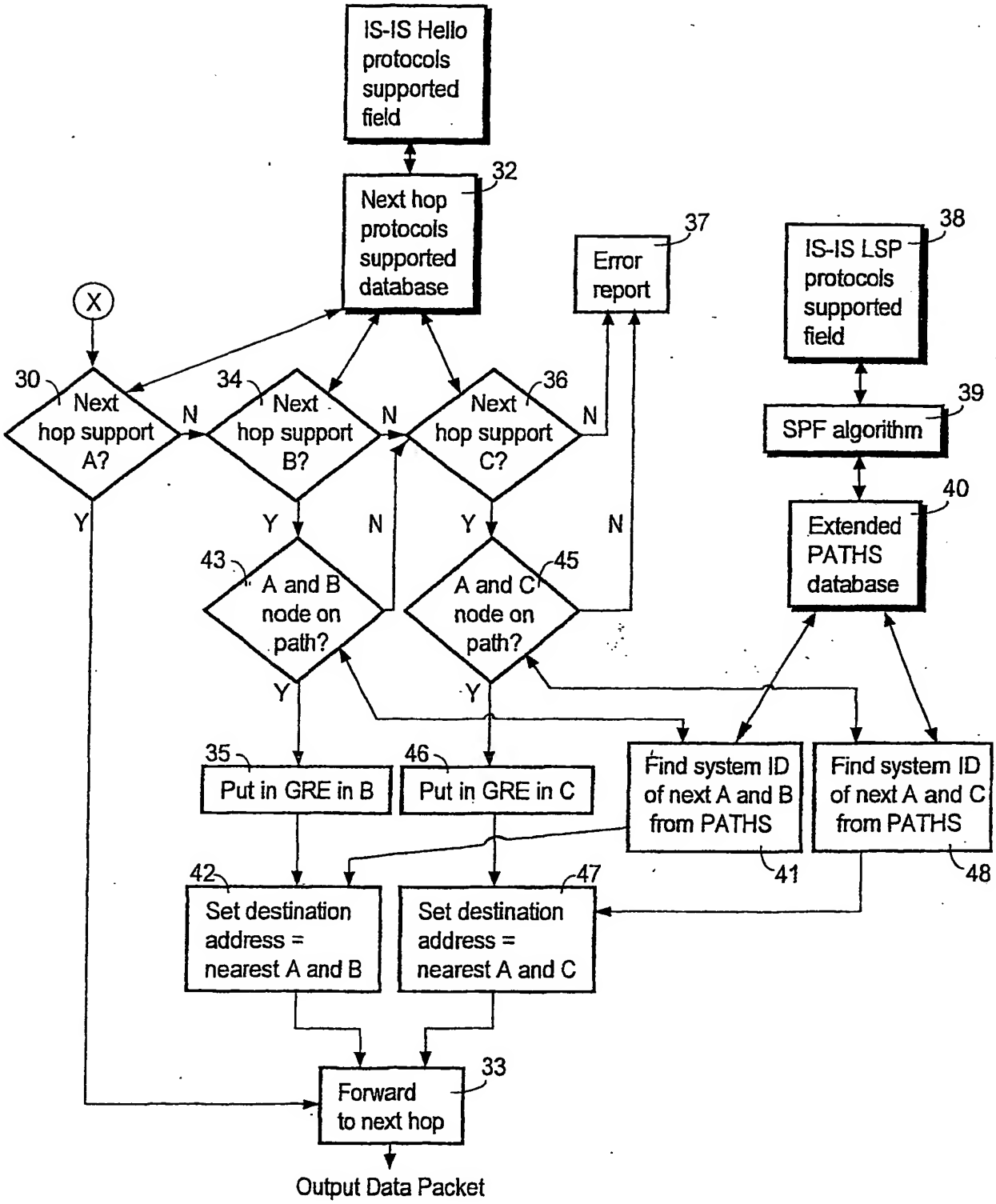

IS-IS Hello protocols supported field
Next hop protocols supported database
32
Error report
37
IS-IS LSP protocols supported field
38
SPF algorithm
39
Extended PATHS database
40
X
30
Next hop support A?
34
Next hop support B?
36
Next hop support C?
N
Y
43
A and B node on path?
45
A and C node on path?
35
Put in GRE in B
46
Put in GRE in C
Find system ID of next A and B from PATHS
41
Find system ID of next A and C from PATHS
48
42
Set destination address = nearest A and B
47
Set destination address = nearest A and C
33
Forward to next hop
Output Data Packet

Fig. 3

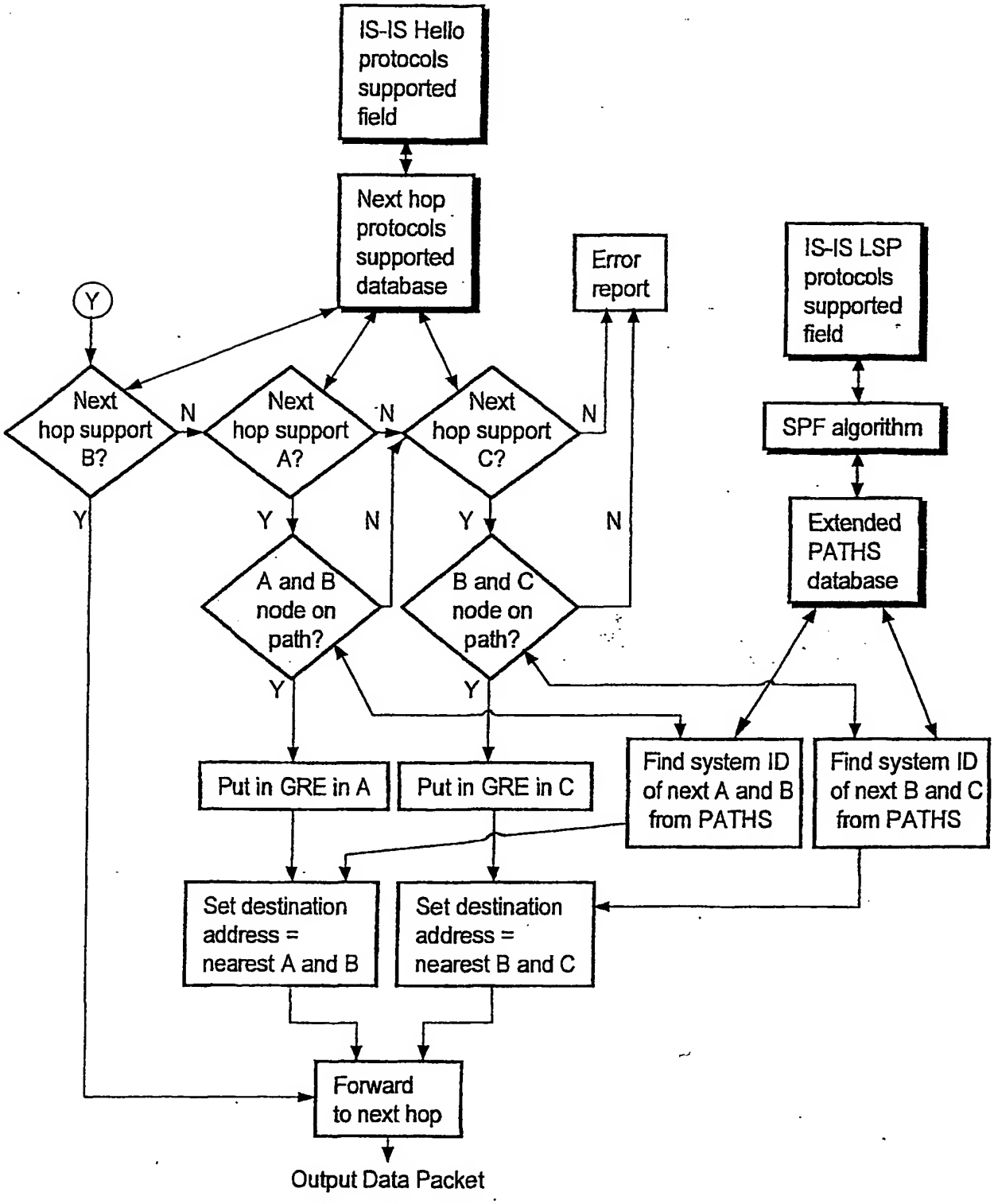

IS-IS Hello protocols supported field
Next hop protocols supported database
Error report
IS-IS LSP protocols supported field
SPF algorithm
Extended PATHS database
Y
Next hop support B?
N
Next hop support A?
N
Next hop support C?
N
Y
Y
N
Y
N
A and B node on path?
B and C node on path?
Y
Y
Put in GRE in A
Put in GRE in C
Find system ID of next A and B from PATHS
Find system ID of next B and C from PATHS
Set destination address = nearest A and B
Set destination address = nearest B and C
Forward to next hop
Output Data Packet

Fig. 4

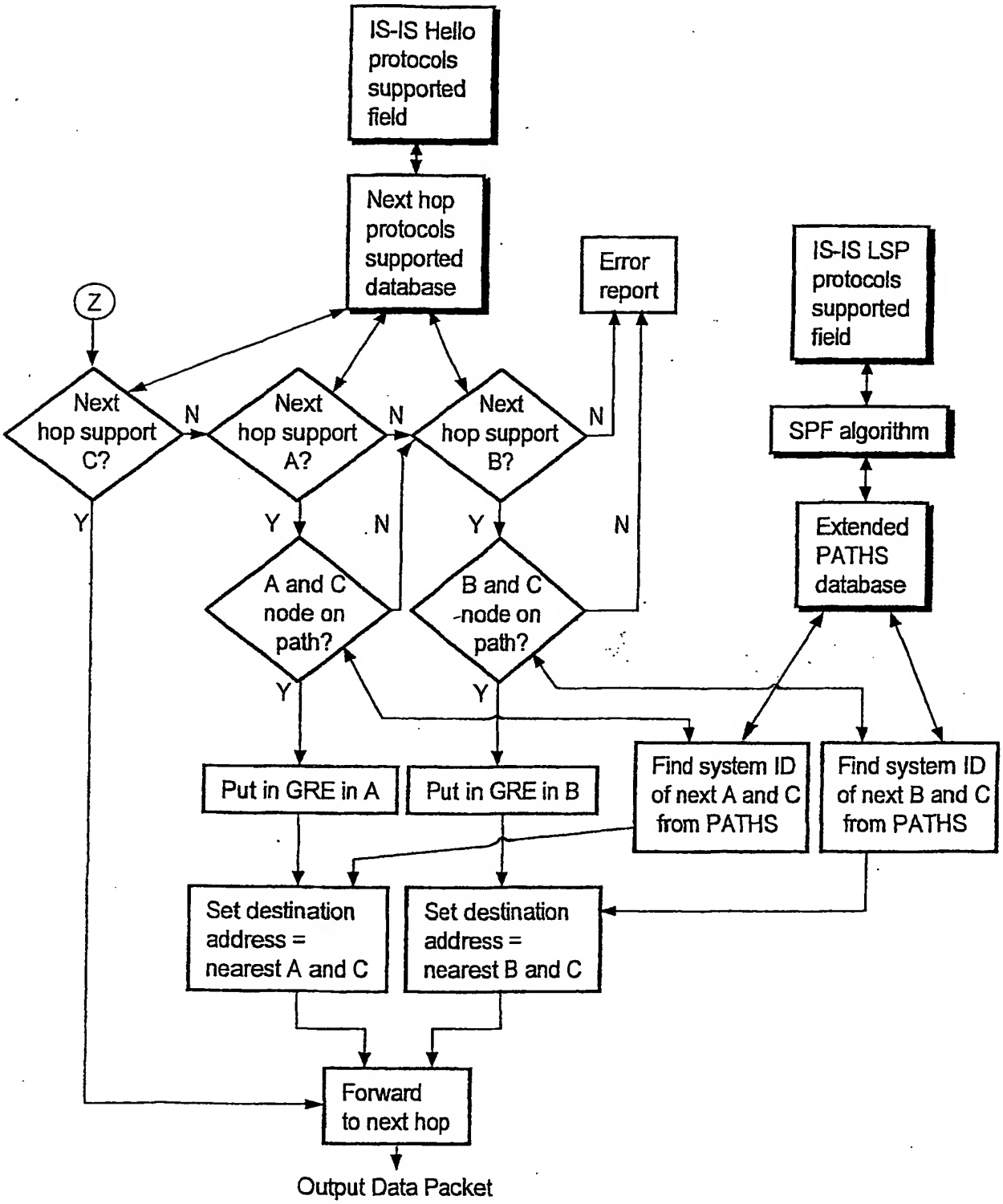

IS-IS Hello protocols supported field
Next hop protocols supported database
Error report
IS-IS LSP protocols supported field
SPF algorithm
Extended PATHS database
Z
Next hop support C?
Next hop support A?
Next hop support B?
N
Y
A and C node on path?
B and C -node on path?
Put in GRE in A
Put in GRE in B
Find system ID of next A and C from PATHS
Find system ID of next B and C from PATHS
Set destination address = nearest A and C
Set destination address = nearest B and C
Forward to next hop
Output Data Packet

Fig. 5

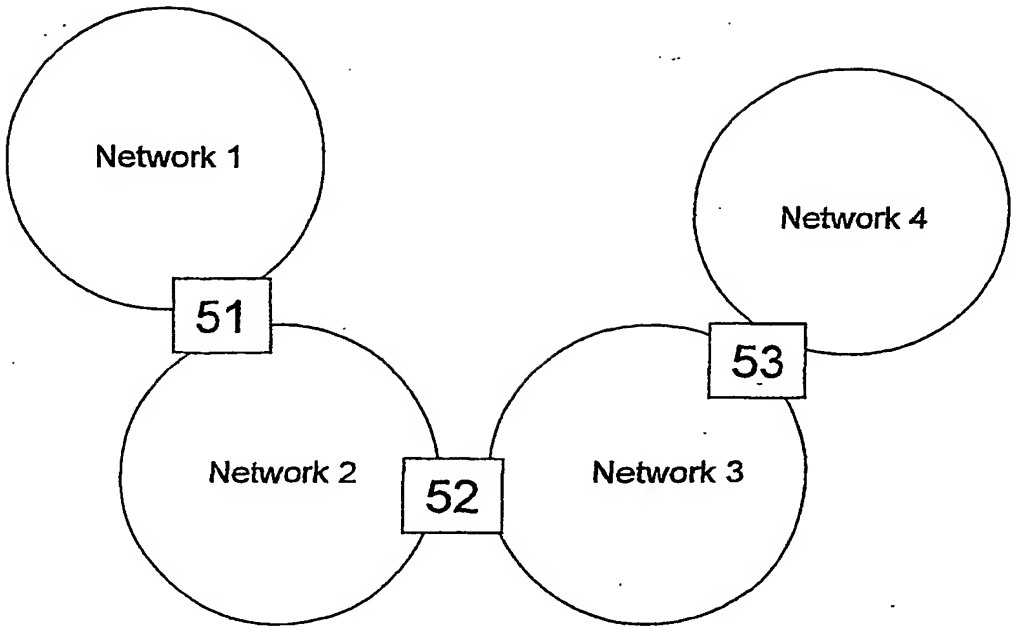

Network 1
Network 4
51
53
Network 2
52
Network 3

Fig. 6

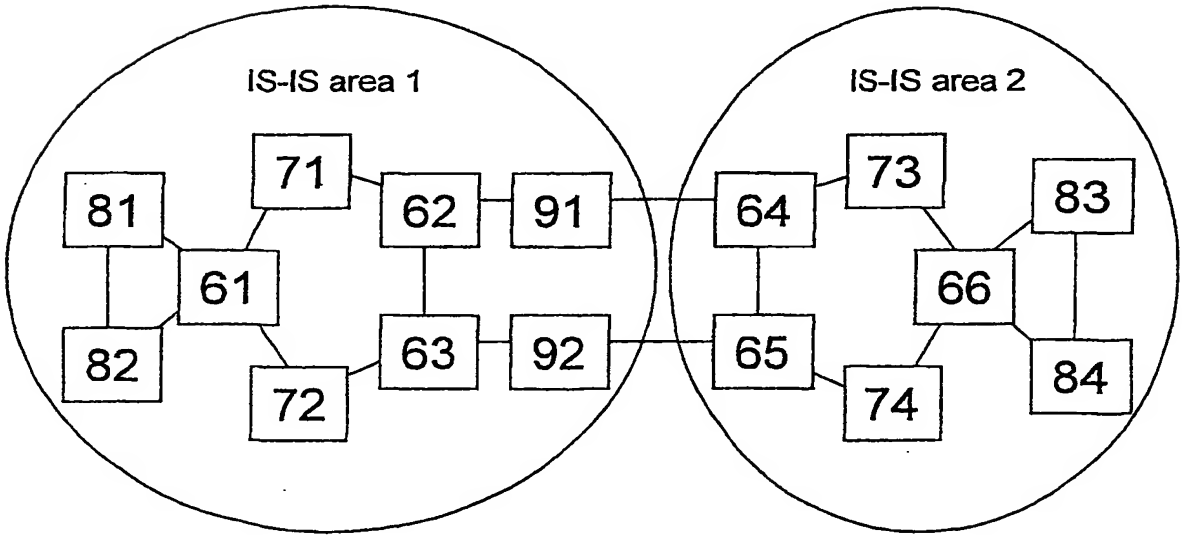

IS-IS area 1
IS-IS area 2
71
81
62
91
64
73
83
61
66
82
63
92
65
84
72
74

Fig. 7